IPEA
EPO
D-80298 Munich
Germany

10 May 2005

Dear Sirs

PCT/G2004/002471
Our ref: Con Serv Broker (PCT)

Thank you for the Written Opinion of the ISA.

The Written Opinion cites 2 category X documents against the independent claims:

D1 Engelstad "Service Discovery and name resolution architectures for on-demand MANETs"

D2 Agrawar "Vinci: a service-oriented architecture for rapid development of web applications"

In light of the citations, the applicant files replacement pages as follows:

Replacement pages 16, 17 and 18 to replace pages 16, 17 and 18 as originally filed.

Triplicate copies will follow by post, together with one set marked to show all changes.

Amended Claim 1 now reads:

1. A method of enabling a client, running on a first computing device that is connected to a second computing device, to use a service on that second computing device, the method comprising the steps of:

Origin Limited
Twisden Works, Twisden Road, London NW5 1DN Tel: +44 (0)20 7424 1950 Fax: +44 (0)20 7209 0643
E-mail: peter.langley@origin.co.uk www.origin.co.uk Registration no. 2211999

(a) a service installed on the second computing device registering its published name, with a service broker on that second computing device;

(b) the client sending a message to the service broker specifying the name of the service;

wherein the published name of the service conforms to a structured naming convention that uniquely identifies the service as a service from a particular vendor, but without specifying the connection point address of that service, to enable the service broker to start up a service without the risk of a clash.

The new text is shown highlighted.

Argument

D1 and D2 are concerned with service discovery in the context of a network of multiple nodes (or computing devices), where a node may offer a service (such as a network printer) and other nodes want to find a node offering the service. In this context, the hard part is to devise an efficient way of locating a node (e.g. IP address) that offers a required service, especially if the network is prone to change. This is not the problem that the present invention is trying to solve, as the amended Claim 1 clarifies.

The present invention is concerned with more localised problems. First, it *assumes* that an external client has *already* connected to the node that can offer a service and the client wants to use that service. Implicitly, the client therefore already knows the address of the node that offers the service it wants to use. (This follows also from original Claim 3, which states that the service broker uses a single well-known port number address so that the client needs only this well known port number to send a message to the service broker. This would be true only if the client already knows the address of the second computing device that the service is installed on).

The localised problems that the present invention deals with are, first, how do you convert service names to local addresses (i.e. port numbers) without the risk of clashes. Secondly, how do you start up services on demand without the risk of conflicts: for performance reasons, it is preferable not to have all the services started up at boot time.

Neither D1 nor D2 address these problems. Hence, in neither D1 nor D2 is there a client already connected to a node and simply wishing to use a service offered by that node without the risk of clashes. Instead, D1 and D2 deal with the problem of determining what nodes might actually offer a given service. This distinction is made explicit in the revised preamble to Claim 1.

The proposed solution further requires, as defined in Claim 1, that the:

> "published name of the service conforms to a structured naming convention that uniquely identifies the service as a service from a particular vendor, but without specifying the connection point address of that service, to enable the service broker to start up a service without the risk of a conflict."

This differs over D1 and D2 for the following reasons. First, in D1 services are identified by a *generic* name:

> "Due to the success of DNS, and the fact that consensus on a universal mechanism for service discovery seems not to be reached within a foreseeable future, DNS SRV records have been standardised to bundle simple service discovery into DNS. They allow a DNS Resolver to resolve a *generic* service name (e.g. of a printer) into an IP address and a port number.....In this paper, we propose to do the same as in the fixed Internet, i.e. to integrate service discovery into the name resolution mechanism... column 2 lines, 1st and 2nd paras.

Because D1 explicitly envisages the use of a *generic* service name, it cannot be said to disclose or suggest the use of a service names that "*conforms to a structured naming convention that uniquely identifies the service as a service from a particular vendor*". Because of this, if services from different vendors use the same generic name and also the same port number, a clash will occur in D1 when a service is started. But with the present invention, that will not happen.

D2 suffers the same deficiencies as D1:

> "Clients can ask VNS for a particular service in one of two ways: by name or by specifying an XPath expression which is applied to the meta data for a service" page 528, left hand column, 10 lines down.

Again, there is no disclosure or suggestion of using a *structured naming convention to uniquely identify the service as a service from a particular vendor.* Hence, if the same name is chosen for a

service from different vendors, or they happen to have the same XPath, then port conflicts can in principle arise.

Support

The revised preamble:

> "A method of enabling a client, running on a first computing device that is connected to a second computing device, to use a service on that second computing device"

finds support at:

> "When an external client, connected to the computing device that has a service broker, wants to use a service on that computing device, it sends a message to the service broker using the well known port number." Page 3 lines 6 – 8.

For the revised characterising portion:

> "wherein the published name of the service conforms to a structured naming convention that uniquely identifies the service as a service from a particular vendor, but without specifying the connection point address of that service, to enable the service broker to start up a service without the risk of a clash."

finds support as follows:

First, the prior art section defines the problem in terms of clashes between services provided by different vendors as follows:

> "However, Independent Software Vendors (ISVs) cannot reserve these port numbers and so the conventional approach makes it difficult for ISVs to create new services since they face the risk of port number allocations being duplicated, with the risk of clashes arising." Page 1 lines 27 – 30.

The solution is defined as:

> "Because service names are made unique by using a structured (and preferably standard and open) naming convention, e.g. by pre-pending the service name with reversed domain information, new connectivity services can be added to devices without having to change the existing configuration of the device. Address clashes are avoided as long as new services use the service broker and a consistent naming convention." Page 2 lines 13 - 17.

It is also clearly the service broker that supplies the connection point address to the external client:

> "The service obtains a connection point by some means and informs the service broker of the connection point address (for TCP/IP this would be a port number, other transport mechanisms (e.g. Bluetooth, serial, USB, IrDA etc.) would have other addressing mechanisms). The service broker then informs the external client of the connection point address of the service." Page 2 lines 20 – 24.

Vendor specific examples of the reverse domain information are given at page 3 lines 24 to page 4 line 5:

> "SymbianOS from Symbian Limited deploys an implementation of the present invention. Examples of service names, originating from within Symbian Limited, that conform to the structured naming convention include the following:
>
> com.symbian.scrfs; where 'scrf' is the Symbian Connect Remote Filing System.
> com.symbian.swinstall - the remote software install service
> com.symbian.syncmlinit - the syncML initiation service
>
> An example of a third-party service might be
> uk.co.ian_mcdowall.pim for a PIM interface service owned by ian_mcdowall.co.uk
> or
> com.big_company.sales_manage for a sales management service provided by
> big_company.com"

In the light of the above arguments and amendments, re-consideration of the present application is requested. Should the examiner require further clarification, a further Written Opinion is requested.

Yours faithfully,

Peter Langley